Jurnal Ilmiah MATRIK Vol.21 No.1 April 2019
ISSN : 1411-1624
e-ISSN: 2621-8089

RANCANG BANGUN APLIKASI PENGENALAN WAJAH UNTUK PASSWORD FILE DENGAN METODE JARINGAN SYARAF TIRUAN

Genrawan Hoendarto[1], Vicni Iskandar[2]
Sekolah Tinggi Manajemen Informatika dan Komputer Widya Dharma
Jalan H.O.S. Cokroaminoto No.445 Pontianak
Sur-el : genrawan@yahoo.com[1], miss.vcn@gmail.com[2]

Abstract : *Data security for computer users is increasingly becoming a concern because it is increasingly vulnerable to illegal access even though the file has been protected with a password. This is possible with the increasing number of applications aimed at hacking owner protection. Artificial neural network that was appointed in this study is one part of computer vision, which in this study is intended to make computers able to "see" through a webcam and recognize that face has access rights to the selected file. So that computers can distinguish facial images, it needs to be trained by applying the back propagation method. The reason for choosing facial recognition is because each person has a different face, so that it can be a more effective security key than conventional methods of making or accessing files that are on a computer.*

Keywords: *Face recognition, artificial neural networks, passwords, back propagation*

Abstrak : *Keamanan data bagi pengguna komputer semakin menjadi perhatian karena semakin rentan terhadap akses ilegal walaupun file sudah diproteksi dengan password. Hal ini dimungkinkan dengan semakin banyaknya aplikasi yang bertujuan untuk meretas proteksi pemilik. Jaringan syaraf tiruan yang diangkat dalam penelitian ini merupakan salah satu bagian dari computer vision, dimana dalam penelitian ini dimaksudkan untuk membuat komputer dapat "melihat" melalui webcam dan mengenali wajah tersebut apakah memiliki hak akses atas file yang dipilih. Agar komputer dapat membedakan citra wajah, maka perlu dilatih dengan menerapkan metode propagasi balik. Alasan dipilihnya pengenalan wajah karena tiap orang mempunyai wajah yang berbeda-beda, sehingga dapat menjadi kunci pengaman yang lebih efektif dibandingkan cara-cara konvensional dalam membuat ataupun mengakses file yang berada dalam komputer.*

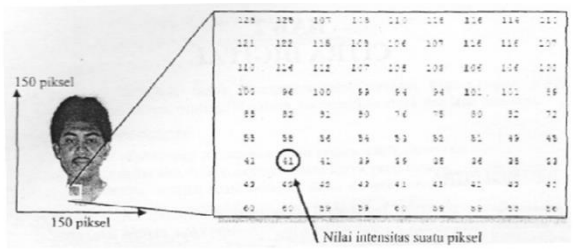
Kata kunci*: Pengenalan wajah, jaringan syaraf tiruan, password, propagasi balik*

1 PENDAHULUAN

Perkembangan ilmu pengetahuan yang begitu pesat, termasuk bidang teknologi informasi dimana penggunaan komputer sudah sangat meluas. Seiring dengan meluasnya pemakaian komputer ini juga menimbulkan rentannya keamanan data. File yang tidak terproteksi dengan baik dapat dengan mudah diakses pihak yang tidak berkepentingan. Pemberian *password* konvensional tidak lagi aman karena banyaknya aplikasi yang dapat digunakan untuk menerobos *password.* Maka harus dicari solusi *password* yang unik yang relatif sulit untuk dibuka. Salah satu yang dapat digunakan adalah salah satu bagian tubuh manusia, yaitu wajah yang pasti berbeda antara tiap orang. *Webcam* yang sudah terintegrasi pada laptop jaman sekarang dapat digunakan untuk menangkap wajah seseorang dalam mengamankan *file* yang penting. Metode jaringan syaraf tiruan dapat digunakan untuk pengenalan wajah dengan menerapkan algoritma propagasi balik. Jaringan syaraf tiruan disusun

ISSN : 1411-1624
e-ISSN: 2621-8089

dengan asumsi yang sama seperti jaringan syaraf biologis, yaitu pengolahan informasi yang terjadi pada elemen pemrosesan (neuron), sinyal antara dua buah neuron diteruskan melalui link-link koneksi, setiap link koneksi memiliki bobot yang terasosiasi, dan setiap neuron menerapkan sebuah fungsi aktivasi terhadap input jaringan dengan tujuan untuk menentukan sinyal.Citra wajah akan direkam dengan webcam dan disimpan sebagai referensi pencocokan saat file akan dibuka [1]. Citra digital adalah citra yang dapat diolah oleh komputer [2]. Perhatikan gambar 1 Sebuah citra grayscale ukura 150x150 piksel (elemen terkecil dari sebuah citra) diambil sebagian (kotak kecil) berukuran 9x9 piksel. Maka, monitor akan menampilkan sebuah kotak kecil. Namun, yang disimpan dalam memori komputer hanyalah angka-angka yang menunjukkan besar intensitas pada masing-masing piksel tersebut."



Gambar 1. Citra grayscale ukuran 150X150 piksel

Jaringan syaraf merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk menstimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran [3].

Jaringan syaraf tiruan menjadi sebuah teknik atau pendekatan yang sangat populer saat ini, khususnya untuk aplikasi suatu pendekatan berbasis citra seperti yang diangkat dalam penelitian ini. Jaringan saraf tiruan berkaitan dengan komputer vision, yaitu sebuah disiplin ilmu yang mempelajari proses menyusun deskripsi tentang objek yang terkandung pada suatu gambar atau mengenali objek yang ada pada gambar. Komputer vision berusaha menerjemahkan citra menjadi deskripsi atau suatu informasi yang merepresentasikan citra tersebut. Jadi *input*-nya berupa citra, sedangkan *output*-nya berupa informasi [2]. *Computer Vision* lebih dari hanya sekedar image recognition [4].

Pada jaringan syaraf tiruan terdiri dari neuron-neuron layaknya neuron yang ada pada jaringan syaraf biologis otak manusia. Informasi (disebut dengan: *input*) akan dikirim ke neuron dengan bobot kedatangan tertentu [3]. Input ini akan diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap neuron. Apabila input tersebut melewati suatu nilai ambang tertentu, maka neuron tersebut tidak akan diaktifkan. Apabila neuron tersebut diaktifkan, maka neuron tersebut akan mengirimkan output melalui bobot-bobot *output*-nya ke semua neuron yang berhubungan dengannya. Demikian seterusnya. Pada jaringan syaraf tiruan, neuron-neuron akan dikumpulkan dalam suatu lapisan tertentu yang disebut dengan lapisan *input* atau lapisan neuron. Neuron-neuron pada satu lapisan akan dihubungkan dengan lapisan yang berada sebelum dan sesudahnya (kecuali lapisan *input* dan lapisan *output*). Informasi yang

ISSN : 1411-1624
e-ISSN: 2621-8089

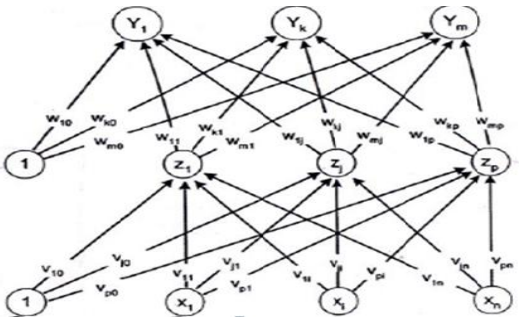
diberikan pada jaringan syaraf tiruan akan dirambatkan lapisan ke lapisan, mulai dari lapisan *input* sampai ke lapisan *output* melalui lapisan lainnya yang dikenal dengan lapisan tersembunyi (*hidden layers*), tergantung pada algoritma pembelajarannya. Berdasarkan uraian diatas, maka penelitian ini bertujuan untuk menghasilkan suatu aplikasi yang dapat melindungi suatu file menggunakan password dengan citra wajah pemilik yang direkam menggunakan *webcam.*

2 METODOLOGI PENELITIAN

Penulis menggunakan desain penelitian hubungan kausal yang memerlukan minimal dua variabel untuk menggambarkan hubungan sebab akibat dan dapat dibuktikan keberlakuan/ketidak-berlakuannya, dengan melakukan percobaan dan dan pengujian terhadap aplikasi yang dibuat dan juga mempelajari literatur-literatur yang berhubungan dengan model jaringan syaraf tiruan dan algoritma propagasi balik serta materi lain yang berhubungan dengan penulisan ini [5].

2.1. Metode Propagasi Balik

Metode propagasi balik merupakan metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks. Metode ini merupakan jaringan syaraf yang popular [6].



Gambar 2. Arsitektur backpropagation

Gambar 2 merupakan arsitektur *backpropgation* dengan *n* buah masukan (ditambah sebuah bias), sebuah layar tersembunyi yang terdiri dari *p* unit (ditambah sebuah bias), serta *m* buah unit keluaran. V_{ij} merupakan bobot garis dari unit masukan x_i ke layar tersembunyi z_j (v_{j0} merupakan bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi z_j). w_{kj} merupakan bobot dari unit tersembunyi z_j ke unit keluaran y_k (w_{k0} merupakan bobot dari bias di layar tersembunyi ke unit keluaran z_k) [6].

Pelatihan *backpropagation* meliputi tiga fase. [6], Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari layar masukan hingga layar keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur dimana selisih antar keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di layar keluaran. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi.

1. Fase I: Propagasi maju

Selama fase ini, sinyal masukan (=x_i) dipropagasikan ke layar tersembunyi menggunakan fungsi aktivasi yang ditentukan. Keluaran dari setiap unit layar tersembunyi (=z_j) tersebut selanjutnya dipropagasikan maju lagi ke layar tersembunyi di atasnya menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan keluaran jaringan (=y_k). kemudian keluaran jaringan dibandingkan dengn target yang harus dicapai (=t_k). Selisih antara t_k - y_k adalah kesalahan yang terjadi. Jika kesalahan ini lebih kecil dari batas toleransi yang ditentukan, maka iterasi dihentikan, namun apabila kesalahan masih lebih

ISSN : 1411-1624
e-ISSN: 2621-8089

besar dari batas toleransinya, maka bobot setiap garis dalam jaringan akan dimodifikasi untuk mengurangi kesalahan yang terjadi.

2. Fase II : Propagasi mundur

Berdasarkan kesalahan t_k - y_k, dihitung faktor δ_k (k=1,2,3,…,m) yang dipakai untuk mendistribusikan kesalahan di unit y_k ke semua unit tersembunyi yang terhubung langsung dengan y_k. δ_k juga digunakan untuk mengubah bobot garis yang berhubungan langsung dengan unit keluaran. Dengan cara yang sama, dihitung faktor δ_k di setiap unit di layar tersembunyi sebagai dasar perubahan bobot semua garis yang berasal dari unit tersembunyi di layar di bawahnya. Demikian seterusnya hingga faktor δ di unit tersembunyi yang berhubungan langsung dengan unit masukan dihitung.

3. Fase III : Perubahan bobot

Setelah semua faktor δ dihitung, bobot semua garis dimodifikasi secara bersamaan. Perubahan bobot suatu garis didasarkan atas faktor δ neuron di atasnya.

Ketiga fase tersebut berulang-ulang terus hingga kondisi penghentian terpenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil daripada batas toleransi yang diijinkan. Algoritma untuk jaringan dengan satu layar tersembunyi (dengan fungsi aktivasi sigmoid biner) adalah sebagai berikut:

Langkah 0: inisialisasi semua bobot dengan bilangan acak kecil.

Langkah 1: jika kondisi penghentian belum terpenuhi, lakukan langkah 2-9.

Langkah 2 : untuk setiap pasang data pelatihan, lakukan langkah 3-8.

Fase I : Propagasi maju

Langkah 3 : tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.

Langkah 4 : hitung semua keluaran di unit tersembunyi z_j (j=1,2,…,p)

$$\boldsymbol{z\_net_j = v_{j0} + \sum_{i=1}^{n} x_i v_{ji}} \quad (1)$$

$$\boldsymbol{z_j = f\left(z_{net_j}\right) = \frac{1}{1+e^{-z\_net_j}}} \quad (2)$$

Langkah 5 : hitung semua keluaran jaringan di unit y_k (k=1,2,…,m)

$$\boldsymbol{y\_net_k = w_{k0} + \sum_{j=1}^{p} z_j w_{kj}} \quad (3)$$

$$\boldsymbol{y_k = f(y\_net_k) = \frac{1}{1+e^{-y\_net_k}}} \quad (4)$$

Fase II : Propagasi mundur

Langkah 6 : hitung faktor δ unit keluaran berdasarkan kesalahan di setiap unit keluaran y_k (k=1,2,…,m)

$$\boldsymbol{\delta_k = (t_k - y_k) f'(y\_net_k) = (t_k - y_k) y_k (1 - y_k)} \quad (5)$$

δ_k merupakan unit kesalahan yang akan dipakai dalam perubahan bobot layar di bawahnya (langkah 7)

Hitung suku perubahan bobok wk_j (yang akan dipakai nanti untuk merubah bobot wk_j) dengan laju percepatan α

$$\boldsymbol{\Delta w_{kj} = \propto \delta_k z_j} \; ; k=1,2,\ldots,m \; ; j=0,1,\ldots,p \quad (6)$$

Langkah 7 : hitung faktor δ unit tersembunyi berdasarkan kesalahan di setiap unit tersembunyi z_j (j=1,2,…,p)

$$\boldsymbol{\delta\_net_j = \sum_{k=1}^{m} \delta_k w_{kj}} \quad (7)$$

Faktor δ unit tersembunyi :

$$\boldsymbol{\delta_j = \delta\_net_j f'\left(z_{net_j}\right) = \delta_{net_j} z_j (1 - z_j)} \quad (8)$$

Hitung suku perubahan bobot v_{ji} (yang akan dipakai nanti untuk merubah bobot v_{ji})

ISSN : 1411-1624
e-ISSN: 2621-8089

$\Delta v_{ji} = \alpha \delta_j x_i$ j=1,2,…,p ; i=0,1,…,n (9)

Fase III : Perubahan Bobot

Langkah 8 : hitung semua perubahan bobot

Perubahan bobot garis yang menuju ke unit keluaran: w_{kj}(baru) = w_{kj}(lama) + Δw_{kj} w_{kj} (k=1,2,…,m ; j=0,1,…,p)

Perubahan bobot garis yang menuju ke unit tersembunyi: v_{ji}(baru) = v_{ji}(lama) + $\Delta v_{ji} w_{kj}$ (k=1,2,…,m ; j=0,1,…,p)

Langkah 9 : pelatihan dihentikan

Setelah pelatihan selesai, jaringan dapat digunakan untuk pengenalan pola. Dalam hal ini, hanya propagasi maju saja yang digunakan untuk menentukan keluaran jaringan.

3. HASIL DAN PEMBAHASAN

Keamanan data yang disimpan dalam suatu media berupa file penting untuk diperhatikan karena merupakan suatu rahasia seseorang atupun suatu institusi. Cara klasik dengan memberikan password sudah tidak efektif lagi karena sudah tersedianya aplikasi-aplikasi yang dapat menembusnya. Ada beberapa alternatif yang dapat dipilih untuk mengamankan file. Cara pertama adalah kryptografi, yaitu dengan melakukan enkripsi pada file dengan berbagai metode yang tersedia. Tetapi cara ini juga kurang terjamin karena tersedianya aplikasi yang dapat menterjemahkan isi file tanpa perlu didekripsi dengan kuncinya, atau dihapus orang karena dianggap file yang rusak. Cara lainnya adalah steganografi, yaitu dengan menyisipkan file ke dalam file lain yang berupa video, gambar ataupun audio. Kelemahannya adalah file yang disisipkan dapat hilang jika file tumpangannya dikompres. Selain itu butuh file media yang besar jika file yang akan disisipkan besar sehingga membutuhkan memori yang besar sewaktu membukanya.

Solusi yang ditawarkan penulis tetap menggunakan password, tetapi tidak menggunakan password berupa karakter yang mudah ditembus, karena menggunakan password yang unik, yaitu menggunakan wajah seseorang yang tentunya tidak ada sama walaupun kembar. Salah satu algoritma yang dapat digunakan dalam membantu komputer mengenali wajah seseorang adalah algoritma propagasi balik (*back propagation*). Algoritma ini akan menganalisa tiap pixel wajah yang ditangkap komputer apakah sesuai dengan data wajah pemilik file yang telah direkam dan disimpan sebelumnya. Maka penulis bermaksud mengembangkan suatu aplikasi yang diberi nama keyFACE yang memungkinkan pengguna menggunakan wajahnya sebagai password hak akses ke dalam aplikasi dan membuka file yang diproteksi. Jenis file yang dapat digunakan aplikasi ini sementara adalah file yang berekstensi doc, xls dan ppt. Pemilik file memasukkan gambar wajahnya dengan menggunakan *webcam* ke dalam aplikasi dan akan dibandingkan dengan rekaman wajah sebelumnya. Jika dikenali sebagai pemilik file, maka aplikasi akan membuka file terproteksi. Untuk meningkatkan keamanan data, maka dapat digunakan password karakter untuk mempersulit yang mencoba menerobos proteksi file.

3.1 Metode Pengenalan Wajah

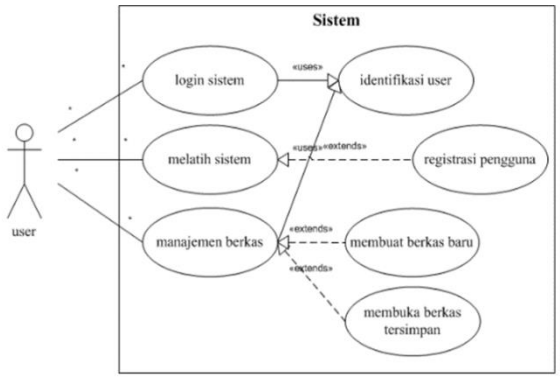
Pengenalan pola pada jaringan syaraf tiruan mengadopsi cara kerja otak manusia

ISSN : 1411-1624
e-ISSN: 2621-8089

dengan membuat aturan tertentu dan data statistik sebagai dasar pengambilan keputusan. Agar sistem menjadi cerdas maka harus dilatih dalam jangka waktu tertentu agar sistem dapat menambahkan aturan-aturan baru dalam mengenali pola wajah yang dimasukkan.

Tahap pertama sistem akan menangkap citra wajah *user* melalui *webcam* dan diolah menjadi citra biner yang kemudian akan dilakukan perhitungan sesuai algoritma propagasi balik. Tahap selanjutnya sistem akan mencocokan hasil perhitungan dengan data citra wajah yang telah disimpan, jika dianggap cocok, maka *user* dapat menciptakan dokumen yang akan diproteksi ataupun membaca dokumen yang sebelumnya sudah diproteksi.

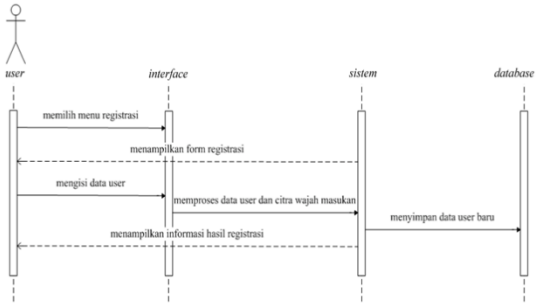
Penulis menggunakan UML untuk menggambarkan kebutuhan perangkat lunak secara visual, yaitu *diagram usecase* dan diagram sekuensial.



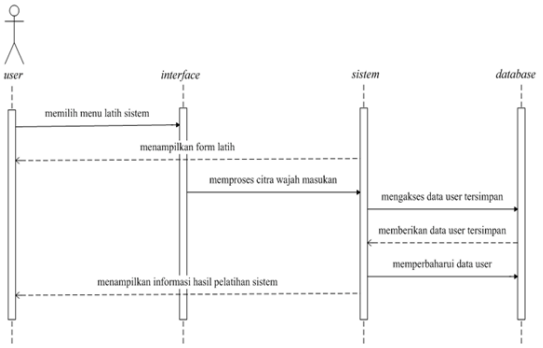
Gambar 3. Diagram use case sistem

Diagram *usecase* terdiri dari seorang *actor* "*user*" dan tujuh *usecase*. *Usecase* "*login sistem*" untuk identifikasi hak akses *user*, usecase "melatih sistem" untuk melatih sistem dengan memasukan citra wajah baru dalam pengenalan pola wajah *user* untuk registrasi pengguna. *Usecase* "manajemen berkas" untuk mengatur dokumen user yg disimpan dalam aplikasi ini. *Usecase* "membuat berkas baru" untuk menciptakan dokumen baru, dan terakhir *usecase* "membuka berkas tersimpan" untuk membuka kembali dokumen yang telah tersimpan.

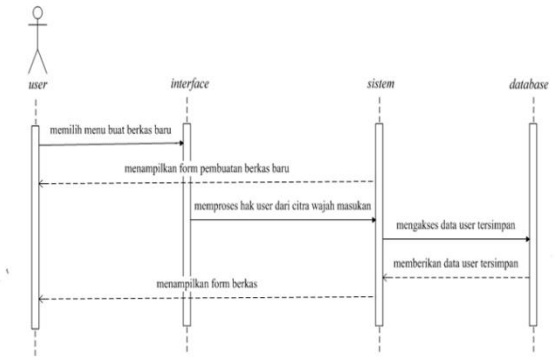
Selanjutnya adalah diagram sekuensial registrasi pengguna, pelatihan sistem dan membuat berkas baru.



Gambar 4. Diagram sekuensial registrasi pengguna



Gambar 5. Diagram sekuensial pelatihan sistem

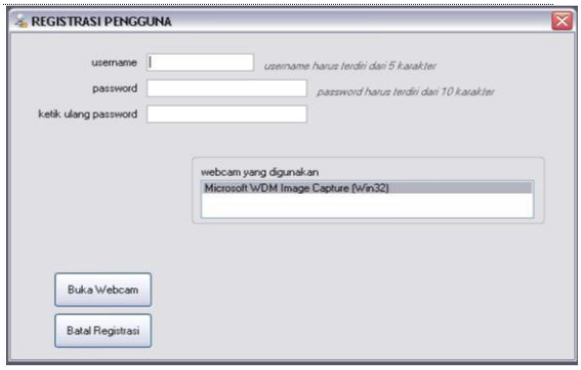


Gambar 6. Diagram sekuensial membuat berkas baru

ISSN : 1411-1624
e-ISSN: 2621-8089

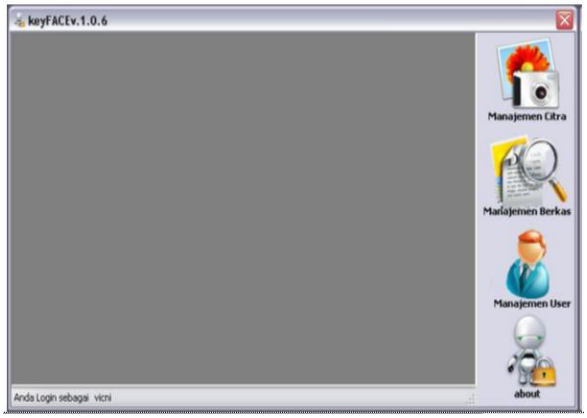
3.2 Pengoperasian Perangkat Lunak

Pertama kali sebelum *user* dapat menggunakan perangkat lunak ini, harus melakukan registrasi memasukkan username dan password yang dapat digunakan sebagai alternatif login sistem jika terjadi kesalahan dalam identifikasi user. Tahap ini memerlukan citra wajah user yang akan ditangkap selama 3 detik, sehingga menghasilkan 3 buah citra wajah yang akan digunakan untuk inisialisasi awal.



Gambar 7. Tampilan menu registrasi user

Setelah regis, maka user dapat melakukan login dengan menggunakan webcam untuk mengambil citra wajah user. Jika tidak ada webcam yang terhubung, maka perangkat lunak ini akan memberikan peringatan. Setelah login sukses, user dapat melakukan manajemen citra yang telah direkam, manajemen berkas dan manajemen user.



Gambar 8. Tampilan menu utama

Pada tahap penciptaan berkas baru, *user* akan diminta untuk memasukkan citra wajah yang akan diidentifikasi berdasarkan jaringan *user* yang tersimpan. Apabila sistem berhasil mengenali citra wajah yang dimasukkan maka sistem akan menciptakan berkas sesuai yang dipilih oleh *user*. Sedangkan pada tahap tampilkan berkas, *user* akan diminta untuk memasukkan citra wajah untuk diidentifikasi apakah *user* tersebut berhak mengakses berkas tersebut. Jika sistem dapat mengenali citra tersebut, maka sistem akan menampikan berkas yang telah dipilih *user*. Untuk tahap pelatihan sistem, *user* akan diminta untuk memasukkan password terlebih dahulu untuk memastikan bahwa citra yang akan diproses benar merupakan *user* yang sedang login dalam sistem. Jika password yang dimasukkan benar, maka sistem akan memproses citra yang dimasukkan, sedangkan jika password yang dimasukkan salah, maka sistem akan memberikan peringatan kepada *user*. Terakhir untuk tahapan penggantian password alternatif, *user* akan diminta memasukkan citra wajah sebagai verifikasi bahwa penggantian password alternatif tersebut benar dilakukan oleh *user* yang bersangkutan. Jika sistem dapat mengenali citra tersebut, maka sistem akan melakukan perubahan password sesuai password baru yang dimasukkan.

3.3 Pengujian Perangkat Lunak

Pengujian dilakukan pada perangkat keras dan perangkat lunak yang sama seperti saat pengimplementasian dengan tujuan menguji proses konversi citra berwarna menjadi citra

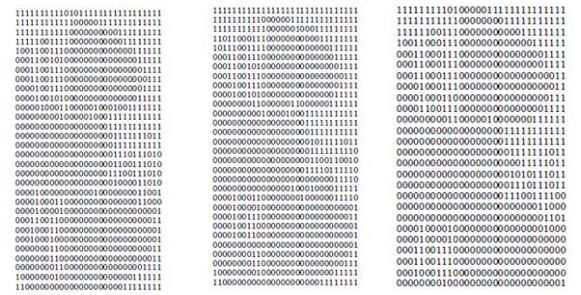
ISSN : 1411-1624
e-ISSN: 2621-8089

biner, menguji proses pelatihan sistem dan menguji kebenaran pengenalan citra dari hasil pelatihan sistem. Data uji pada registrasi sistem berupa citra *user* diambil sebanyak tiga kali untuk pelatihan sistem.

Gambar 9. Citra user1.BMP, user2.BMP dan user3.BMP

Gambar 10. Citra user4, user5 dan user6

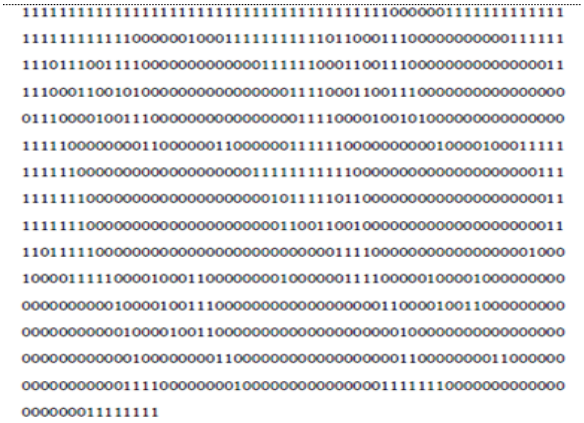
Pengujian proses konversi citra berwarna yang telah ditangkap menggunakan *webcam* menjadi citra biner agar dapat digunakan dalam tahap selanjutnya. Pada pengujian ini, sistem akan mengubah ukuran citra menjadi citra 30x30 pixel. Ini dilakukan untuk mengurangi lama proses dan kerumitan sistem nantinya. Konversi warna ini dilakukan dengan mengambil nilai rata-rata dari nilai RGB pixel tersebut. Apabila nilainya lebih besar dari 128, maka berikut tersebut memiliki nilai biner 0, dan apabila nilainya lebih kecil dari 128 maka pixel tersebut akan memiliki nilai biner 1. Berikut ini adalah citra biner dari tiap data uji:



Gambar 11. Nilai biner dari citra user1, user2 dan user3

Dari hasil pengujian, dapat disimpulkan bahwa keyFACE telah dapat melakukan reduksi ukuran citra dan mengkonversi citra berwarna menjadi citra biner dengan baik.

Setelah sistem berhasil melakukan binerisasi citra *user*, sistem akan mengubah matriks nilai biner yang didapat menjadi sebuah vektor. Nilai pada vektor inilah yang akan menjadi node input awal jaringan. Penentuan jumlah node disesuaikan dengan ukuran matriks 30x30 sehingga menghasilkan 900 buah node input awal. Data yang dimasukkan ke sistem akan sesuai dengan nilai biner yang telah didapatkan, yaitu biner 1 untuk pixel yang berisi citra dan biner 0 untuk pixel yang tidak berisi citra. Sebagai contoh untuk citra user1.BMP yang memiliki matriks biner seperti gambar 12, maka akan menghasilkan vektor input:



Gambar 12. Vektor input dari citra user1

Setelah mendapatkan nilai vektor *input* (*node input*) jaringan, maka sistem akan melakukan inisialisasi awal bobot-bobot garis yang akan digunakan dalam proses pelatihan pengenalan wajah. Bobot-bobot awal jaringan ini ditentukan secara acak, dan angka yang digunakan berkisar dari -0,5 hingga 0,5. Inisialisasi awal untuk bobot garis *input* menuju *layer hidden* menghasilkan sebuah matriks berukuran 900x10. Hal ini disesuaikan dengan

ISSN : 1411-1624
e-ISSN: 2621-8089

jumlah *node input* dan *node hidden* jaringan yang ditentukan. *Node input* jaringan telah ditentukan sebelumnya, yaitu 900. Dan node hidden yang digunakan oleh penulis adalah sebanyak 10 buah node. Berikut ini adalah tabel yang merepresentasikan matriks bobot garis input menuju *node hidden* untuk *user*.

i \ j	1	2	3	4	5	6	7	8	9	10
1	0.24392	-0.3299	-0.2111	0.17796	-0.0377	0.19194	0.44247	-0.1898	-0.4635	0.30647
2	-0.1626	-0.3402	0.13433	0.32288	-0.3913	0.13417	-0.4036	-0.4683	0.44721	0.47348
3	0.26424	-0.2793	-0.4477	0.39169	0.05868	-0.0598	0.3356	-0.033	-0.1924	-0.4812
4	0.37343	-0.2175	-0.1756	0.32921	-0.3895	0.31666	-0.2643	0.30356	-0.1873	-0.4157
5	-0.4401	0.19925	0.27039	0.02688	0.03517	-0.3855	-0.2485	-0.3468	0.19743	0.25394
6	-0.2955	0.43017	0.2086	0.4443	-0.2882	-0.0768	0.43471	-0.1	0.19254	0.18718
7	-0.226	-0.2612	0.35614	-0.0934	0.23512	-0.2527	0.40499	-0.2133	-0.2349	-0.1031
8	-0.0122	0.19363	0.32765	0.10888	-0.0972	0.04553	0.23118	0.27061	0.05071	-0.0472
9	-0.2931	-0.4992	-0.3622	-0.3912	-0.2628	-0.3348	-0.3708	0.43063	0.13325	0.14227
10	0.45208	0.4496	-0.1363	-0.1611	-0.0515	0.4519	-0.3676	0.45571	-0.4214	0.36922
11	-0.2931	0.00312	-0.1258	-0.0806	-0.004	-0.4334	0.14101	0.49704	0.37861	0.32247
12	0.41269	0.31411	0.28116	-0.1965	-0.3467	-0.2094	-0.4937	0.41461	0.21098	0.1951
13	0.31056	0.19826	-0.4796	-0.4645	-0.2671	-0.1497	0.05666	-0.3004	0.17261	-0.0144
14	-0.4623	-0.1544	0.2173	-0.3464	0.37793	0.04288	-0.2787	0.10366	0.39115	0.16266
15	0.1337	-0.1987	-0.2563	0.01846	0.07632	-0.3066	0.15745	-0.2432	0.17206	-0.2761

Gambar 13. Bobot garis node input menuju node hidden (v_{ji})

Jaringan yang dibangun pada keyFACE menggunakan hanya satu *layer hidden* yang berisi 10 *node hidden*. Berikut ini adalah bobot bias tiap *node hidden* dan bobot garis *node hidden* menuju *node output*.

Tabel 1. Bobot bias node hidden (v_{j0})

Bias Hidden ke-	*Bobot Bias*
1	-0.42461
2	-0.24647
3	0.47717
4	-0.42677
5	0.18704
6	-0.04529
7	-0.44559
8	-0.3322
9	-0.36718
10	0.31924

Tabel 2. Bobot garis node hidden menuju node output (w_{kj})

Garis Node Hidden	*Bobot garis*
1	-0.23784
2	0.06184
3	-0.35214
4	0.42231
5	0.39448
6	-0.33614
7	0.26625
8	-0.2105
9	-0.08493
10	0.29147

Untuk target awal jaringan ini, sistem akan mendefinisikan sebuah nilai acak yang kemudian akan menjadi target jaringan. Inisialisasi acak ini akan menjadi ciri dari setiap *user*, sehingga jaringan dapat membedakan *user* yang satu dengan lainnya. Untuk jaringan *user*, sistem menentukan target awal yaitu 0.896178960800171 dengan nilai bias pada lapisan *output* adalah 0.00801. Sesuai dengan kaidah propagasi balik, tahap pertama yang akan dikerjakan oleh sistem adalah tahap propagasi maju. Pada tahap ini semua *node input* akan diteruskan ke lapisan yang ada di atasnya, yaitu lapisan *node hidden*.

Untuk menghitung nilai keluaran di jaringan tersembunyi (*node hidden*) digunakan rumus (1): dimana v_{j0} merupakan *node* bias pada *layer hidden*, x_i merupakan vektor *input* jaringan, dan v_{ji} merupakan bobot tiap garis yang menghubungkan *node input* ke *node* tersembunyi. Nilai *node hidden* merupakan hasil dari aktivasi z_net_j. Aplikasi keyFACE menggunakan fungsi aktivasi sigmoid biner, sehingga nilai *node hidden* (z_j) didapat dengan rumus (2). Setelah didapatkan *node hidden*, bobot pada *node hidden* ini kemudian akan diteruskan lagi ke lapisan selanjutnya. Seperti yang telah dipaparkan, aplikasi keyFACE hanya menggunakan sebuah lapisan tersembunyi. Sehingga lapisan diatas lapisan tersembunyi adalah lapisan keluaran. keyFACE hanya menggunakan sebuah *node output* sebagai unit keluaran. Untuk keluaran pada lapisan *output* digunakan rumus (3): dimana w_{k0} merupakan node bias *output*, z_j merupakan nilai *node hidden* yang telah didapatkan sebelumnya, dan w_{kj} merupakan bobot garis yang menghubungkan *node hidden* dengan *node output*. Nilai *node output* (y_k) merupakan nilai yang dihasilkan dari fungsi aktivasi (4).

ISSN : 1411-1624
e-ISSN: 2621-8089

Nilai *output* yang dihasilkan masing-masing adalah 0.5026 , 0.47349 , dan 0.63421. Dari nilai *output* yang dihasilkan ini kemudian akan dicari nilai minimum error kuadrat (MSE) guna menghasilkan jaringan yang lebih baik. Nilai error yang ditolerir oleh sistem adalah sebesar 0.1. Apabila nilai MSE yang didapat dari tiga node *output* ini lebih kecil daripada nilai *error* yang ditolerir oleh sistem, maka proses pelatihan akan dihentikan. Untuk menghitung nilai MSE digunakan rumus:

$$MSE = \frac{1}{2}\sum(t_k - y_k)^2 \qquad (10)$$

Dimana t_k merupakan *target ouput*, yaitu 0.896178960800171, dan yk merupakan nilai *node output* masing-masing citra. Nilai MSE yang didapatkan adalah 0.201099046194796. Angka ini lebih besar dari batas kesalahan yang ditolerir oleh sistem. Sehingga sistem akan melakukan peng-*update*-an bobot jaringan sehingga didapatkan nilai yang lebih kecil yang ditolerir oleh sistem. Jumlah perulangan (epoh) yang dikerjakan oleh sistem adalah maksimal 100 kali epoh dengan laju pembelajaran (alpha) adalah 0.2. Jika dalam masa perulangan didapatkan nilai MSE yang lebih kecil daripada 0.1 maka perulangan akan dihentikan dan sistem akan menggunakan bobot yang telah didapatkan untuk mengenali *user*.

Pada kasus *user* ini dilakukan perulangan sebanyak empat kali hingga dihasilkan jaringan yang memenuhi standar sistem. Nilai bobot garis *node input* menuju *node hidden*, bobot garis *node hidden* menuju *node output*, dan bobot bias lapisan *hidden* yang baru terlihat pada tabel 3 dan 4.

Tabel 3. Bobot baru garis node hidden menuju node output (w_{kj})

Bias Node Hidden ke-	*Bobot*
1	-0.18528
2	0.07059
3	-0.3465
4	0.46017
5	0.43888
6	-0.24895
7	0.35778
8	-0.11884
9	0.01496
10	0.338255

Tabel 5. Bobot baru bias lapisan hidden (v_{j0})

Bias Node Hidden ke-	*Bobot*
1	-0.43169
2	-0.24594
3	0.47559
4	-0.42275
5	0.19222
6	-0.04654
7	-0.4455
8	-0.33227
9	-0.36742
10	0.31931

Sedangkan nilai bobot baru bias *output*-nya adalah 0.17738. Pengujian kebenaran pengenalan citra dari hasil pelatihan sistem dilakukan dengan cara melakukan login ke dalam sistem. keyFACE akan mulai menangkap citra *user*, dan menganalisa menggunakan kaidah propagasi balik dengan menggunakan tahap propagasi maju, namun dengan menggunakan bobot-bobot jaringan yang telah didapatkan sebelumnya dari tahap pelatihan sistem dari sesi registrasi sistem. Pengujian dilakukan dengan memperhatikan beberapa aspek, yaitu: intensitas cahaya, perubahan mimik wajah, perubahan jarak antara *webcam* dan *user*, perubahan posisi wajah dan penggunaan asesori pada wajah *user* seperti kacamata.

3.4 Evaluasi Perangkat Lunak

Setelah didapatkan hasil pengujian, penulis menemukan beberapa keunggulan dan kelemahan yang dimiliki oleh perangkat lunak keyFACE ini. Keunggulannya: menggunakan citra wajah yang

ISSN : 1411-1624
e-ISSN: 2621-8089

unik sebagai password, menyimpan file dokumen yang diproteksi ke dalam database yang tidak dapat diakses tanpa menggunakan perangkat lunak ini, dan dapat belajar mengenali citra *user* dengan proses latih sistem. Sedangkan kelemahannya adalah: tidak dapat mengenali citra *user* yang menggunakan aksesoris seperti kacamata, sistem tidak dapat mengenali *user* jika jarak yang digunakan *user* relatif jauh berbeda, sistem tidak dapat mengenali *user* jika intensitas cahaya yang ada jauh berbeda dengan saat pelatihan, perubahan mimik wajah *user* yang signifikan tidak dapat dikenali oleh sistem dan sistem tidak dapat mengenali wajah *user* yang posisi wajahnya berbeda dengan citra yang digunakan pada pelatihan.

4. KESIMPULAN

Teknik pengamanan file menggunakan password sudah tidak aman lagi karena dapat dengan mudah dibobol oleh orang lain, maka salah satu alternatif solusinya adalah dengan teknik pengenalan objek, khususnya wajah menggunakan jaringan syaraf tiruan. Algoritma *Back Propagation* merupakan salah satu algoritma yang dapat digunakan untuk mengenali objek dan dinilai lebih baik sebab dapat melakukan pengembangan terhadap jaringan yang telah dibentuk. Algoritma ini sangat bergantung pada data pemasukan data awal dalam mengenali objek, sehingga jika terdapat perubahan yang cukup signifikan pada saat mengenali objek, maka sistem kemungkinan besar tidak dapat mengenalinya. Dalam menerapkan algoritma *Back Propagation* disarankan citra masukan tidak berukuran besar untuk mempersingkat waktu pelatihan sistem, tetapi tetap harus mengandung cukup nilai utamanya. Pengembang aplikasi ke depannya dapat menyisipkan password citra wajah ke dalam file sehingga tidak perlu menggunakan database sebagai penyimpan data file, karena jika database sudah menampung terlalu banyak data file, maka sistem akan menjadi berat dan kinerjanya menurun.

DAFTAR PUSTAKA

[1] Puspitaningrum, Diyah, *Pengantar Jaringan Syaraf Tiruan*, Yoyakarta: Andi, 2006
[2] Sutoyo et al., *Teori Pengolahan Citra Digital*. Yoyakarta: Andi, 2009
[3] Kusumadewi, Sri. 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab*. Yogyakarta: Graha Ilmu.
[4] Fadlisyah dan Sigit Suryantoro. 2007. *Computer Vision & Pengolahan Citra*. Yoyakarta: Andi.
[5] Priyono, *Metode Penelitian Kuantitatif*, Surabaya, Zifatama Publishing, 2016.
[6] Siang, J.J., *Jaringan Syaraf Tiruan dan Pemogramannya Menggunakan MATLAB*, Yoyakarta: Andi, 2009